April 1995

DCIEM No. 93-54

An Expert System Opponent For Wargaming

DTIC
ELECTE
NOV 2 9 1995
S D
G

T.R.H. Cutmore
H.D. Gamble

Defence and Civil Institute of Environmental Medicine
1133 Sheppard Avenue West, P.O. Box 2000
North York, Ontario
Canada M3M 3B9

19951124 024

© HER MAJESTY THE QUEEN IN RIGHT OF CANADA (1995)
as represented by the Minister of National Defence

© SA MAJESTE LA REINE EN DROIT DU CANADA (1995)--
Défense Nationale Canada

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

Abstract

Planning and decision making for battle management are difficult and time-critical tasks. To facilitate the study of these cognitive processes in the laboratory, an army wargame facility (WARFAC) has been developed at DCIEM. This report details the design and implementation of an extension to WARFAC, an expert system opponent. This extension will allow controlled experiments with a single human subject. Initial test results are presented and its possible uses as a research tool are outlined.

Accesion For	
NTIS CRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and / or Special
A-1	

Contents

1 Executive Summary

Directing the control of armed forces in battle is a complex and time-critical task. It is of considerable practical interest to understand the planning and decision making processes that are involved. One of the means for investigating human decision making and the associated planning processes is to devise a simulation environment in which these processes can be made to occur and thereby studied. To this end, an army wargame facility (WARFAC) was devised at DCIEM through the 1980's. WARFAC permits two players to manipulate resources to accomplish a tactical mission. As an extension to WARFAC, an intelligent computer opponent (WFICO) has been developed to provide consistent opposition and thus simplify the task of assessing the human player's performance.

WFICO was designed with a dynamic knowledge base and rules with a layered structure that models the army command and control hierarchy. This report details the status of this expert system, the expert system shell used in its development, and the potential use of WFICO as a research tool.

Initial testing of WFICO used a small knowledge base consisting of simple rules governing combat and movement of units. Results with this simple prototype demonstrated that WFICO could respond reasonably well against an experienced human opponent in the defence of an identified objective (prevent a bridge crossing).

Together with WARFAC, WFICO provides an open ended testbed for research in command and control (C^2). It can be used to support research in the assessment of the C^2 decision process, assist in C^2 decision tasks, or test decision making rules.

2 Introduction

Directing the control of armed forces in battle is a complex and time-critical task. It is of considerable practical interest to understand the planning and decision making processes that are involved. In addition, it is important to determine under what conditions these decision processes might degrade and what might be done to supplement them. One of the means for investigating human decision making and the associated planning processes is to devise a simulation environment in which these processes can be made to occur and thereby studied. Variables of interest may be manipulated and their results observed. This also offers a cost effective method. Consequently, an army wargame facility (WARFAC) was devised at DCIEM through the 1980's (1).

WARFAC is a multi-player, land based tactical wargame that can be run in conjunction with ancilliary cognitive tasks and an electrophysiological data collection and analysis system. Each player is in command of a division of units composed of tanks, infantry, helicopters, etc.. The game has two phases, combat phase, and movement phase. In the combat phase, the players take turns executing attacking and defensive fire. In the movement phase, both players may submit movement orders for their units. The WARFAC system has been assessed by army officers to be capable of providing some of the essential features for the study of army command and control (C^2). However, since each player's activity results from as yet poorly understood decision processes, the combined behaviour of the players can be highly variable. From the experimenter's perspective, this is undesirable since it becomes difficult to draw general conclusions about how either human player is operating.

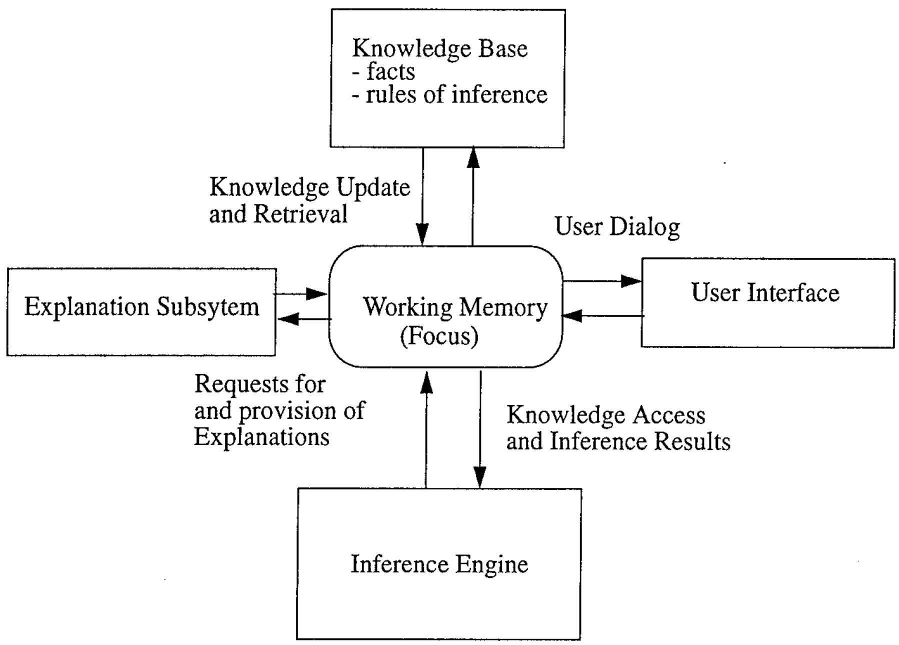
It was therefore deemed desirable to build an intelligent computer opponent (WFICO) which could be defined in terms of a set of rules for game play. Such a system would provide a consistent opponent and thus simplify the task of assessing the human player's performance. It was determined that rule-based methods would provide an effective approach (2). Figure 1 illustrates the general model for a rule-based expert system.

This conclusion was based on the observation that considerable documentation exists for battle procedures that can be interpreted in condition-action format. For example, Defensive Operations (3) and Land Formations in Battle (4) outline defensive maneuvers in sufficient detail that rules may be defined which provide for the deployment of forces. Additionally, rule-based expert systems have the virtue of supporting an incremental and modular approach and good generic expert system shells exist for development of such systems (5). Therefore, we designed an intelligent computer opponent (WFICO) with a dynamic knowledge base, inference engine (2) and multiple levels of rules that model the C^2 hierarchy. This report details the status of this expert system, the shell used in its development, and the potential of the expert system as a research tool.

3 The Intelligent Computer Opponent

3.1 Hardware Requirements

WARFAC runs on a DEC VAX minicomputer. The players sit at Commodore Amiga microcomputers, which provide a graphical user interface to the game based on map displays. The Amigas also provide automated data collection



Expert System Schematic. An expert system (ES) consists of a set of functional units which operate to provide an interactive symbolic problem solving partner to a human. As a dialog the interaction includes mutual requests for information between the human and the ES. In addition, the ES may be requested to explain its reasoning processes. The focus of the dialog is maintained within the ES by a working memory. A knowledge base maintains a set of facts and rules of inference about the domain of expertise. The inference engine contains a set of inferencing control functions (e.g., backward and forward reasoning functions) which access known facts and rules and deduce new ones.

Figure 1:

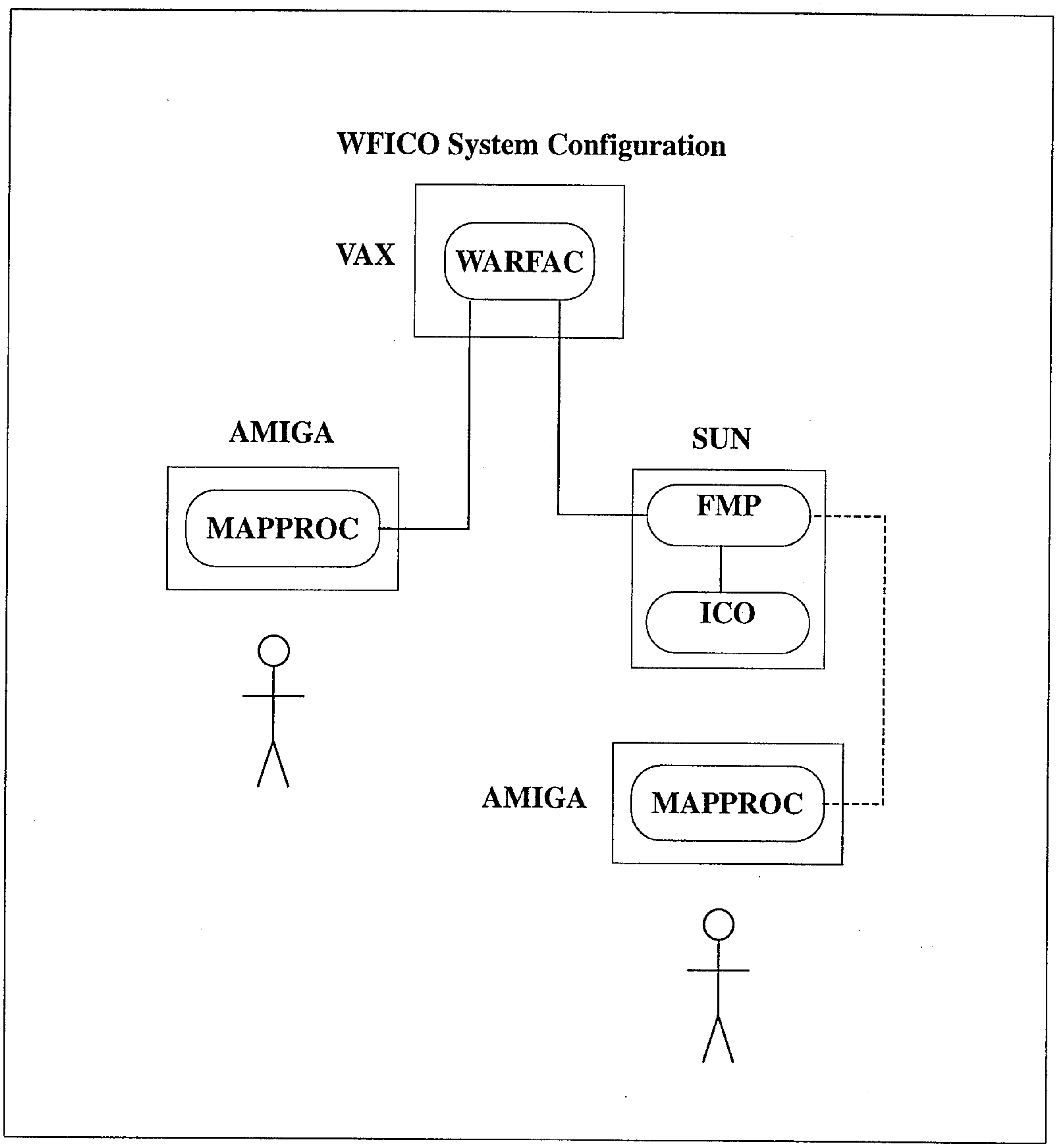


Figure 2: Hardware and Software Configuration. The wargame proper (WARFAC) runs on a VAX, the user Map Processors (MAPPROC) run on Amigas, and the Fake Map Processor (FMP) and Intelligent Computer Opponent (ICO) run on a SUN.

for ancilliary cognitive tasks and electrophysiological recordings, if desired. WFICO runs on a SUN workstation. The VAX, SUN and Amigas are networked together by Ethernet. The logical connections are shown in Figure 2. In the figure, the SUN is shown interposed between the VAX and an Amiga. To the VAX, the SUN running WFICO is indistinguishable from an Amiga with a human player. WFICO can optionally pass game information on to an Amiga with map displays as a debugging aid. In the future, this passthrough capability could be used to provide an expert advisor for a human player (see section 5.2).

3.2 Software Components

A user interfaces with WARFAC through a map processor (MAPPROC) that runs on an Amiga. For WFICO a fake map processor (FMP) was designed to act as the interface between WARFAC and the Intelligent Computer Opponent (ICO). The FMP emulates the Amiga Map Processor interface to the VAX and isolates the ICO from the details of communicating with WARFAC. Within the ICO there is a further distinction between rules that model the C^2 hierarchy and lower level rules specific to WARFAC. The next two sections describe the FMP and ICO in detail.

3.3 The Fake Map Processor (FMP)

The FMP contains C language data structures and code essentially identical to parts of the Map Processor program that runs in the Amiga display stations. It builds a local copy of the information it is sent about the world. While the Amiga map processor presents this information graphically to a human player, the FMP presents it to the ICO in the form of structured data objects. It also contains code to initialize the ICO and to start inferencing. The Amiga map processor accepts commands that the player enters by clicking with a mouse, and sends them to the wargame. The FMP does the same with command messages from the ICO.

The FMP receives messages from the wargame on the VAX, updates its own data structures as well as the ICO's knowledge base, and initiates inferencing by the ICO when appropriate. For example, when the wargame sends a "UNIT_INFO" message to the FMP, the FMP updates the information it has on that specific unit, then updates the ICO's representation of the unit. No inferencing by the ICO is required. When the wargame sends a "MOVE" message to the FMP, the FMP does not have to update any of its data structures, but does initiate inferencing by the ICO for the movement phase. When the FMP receives a command message from the ICO, for example, a list of units to be moved to a new map location, it translates it into the WARFAC message format and passes it on.

The FMP has provision for a pass-through mode where it sends messages from the wargame to both the ICO and an Amiga display station (dashed line in Figure 2). In this mode, commands from the ICO are sent to the display of the SUN only, and not back to the wargame. A human player at the Amiga may then consider the action the ICO would have performed as a recommendation when deciding on an action. With some embellishment of the user interface and explanation facilities, this could serve as the basis for a tactical advisor to a human player (this concept is further discussed in section 5.2).

3.4 Intelligent Computer Opponent

The ICO is a knowledge based system written in the "Nexpert Object" (6) expert system shell. It consists of declarative knowledge, in the form of classes and objects, which represents such things as infantry units and subdomain expert state information, and procedural knowledge, in the form of rules, that express actions such as when a unit should fire. The ICO has a model of the "world" of the game that constitutes its environment, and a hierarchy of command and control that models the flow of orders and information through the chain of command.

3.4.1 Nexpert Object

Nexpert Object is an expert system development environment that runs on many different hardware platforms. It is used at DCIEM on both SUN workstations and Macintosh microcomputers. It provides graphically based development tools, and allows easy integration with components of a system, such as the FMP, that are better suited to implementation in a lower level language, such as C.

Nexpert allows knowledge to be represented as object oriented hierarchies of classes and objects. For example, in the ICO there is a class named "unit" with an instantiated object for each tank, helicopter, etc., in the game. Reasoning is expressed in logical inferencing rules that operate on these objects. The ICO contains rules such as "If a unit is disrupted then it can't move." The Nexpert inferencing engine is very flexible and powerful, allowing multiple inferencing strategies to be used simultaneously. In the interest of simplifying the design and enhancing maintainability, we chose an exclusively backward chaining inferencing paradigm (7) for the ICO.

3.4.2 ICO Command and Control Hierarchy

The ICO is structured as a set of separate subdomain experts that communicate through a chain of command (8). All orders start at the Division level. The Division commander (DIV) passes orders down to a Brigade commander (BGD). The Brigade commander in turn passes orders down to subdomain experts such as Intelligence (INT), Operations (OPS), and Logistics (LOG).

Each subdomain expert consists of a domain specific set of knowledge structures and inferencing rules. These in turn, are built up from a number of lower level structures and rules. At the lowest levels (below the dashed line in Figure 3) there are knowledge structures and rules specific to the wargame.. These pertain to such things as who can fire on whom, when units may fire, and how they fire. These are used by the various subdomain experts such as OPS when carrying out their orders. OPS has rules for performing two types of defensive fire: pre-emptive, where the opponent's offence is anticipated; and reactive, in response to direct attacks by the opponent. These rules send orders to Artillery for indirect fire, and request unit movement required for battle management.

Inferencing follows the C^2 chain of command. For example, when the ICO gets a "move" message from the FMP indicating that the wargame is in the movement phase, the ICO Division commander gives a movement order to the Brigade commander. The Brigade commander gives a movement order to Intelligence and to Operations. Intelligence requests any unit movements required for surveillance purposes. Operations requests any unit movements for battle management. After resolving all movement requests, the Brigade

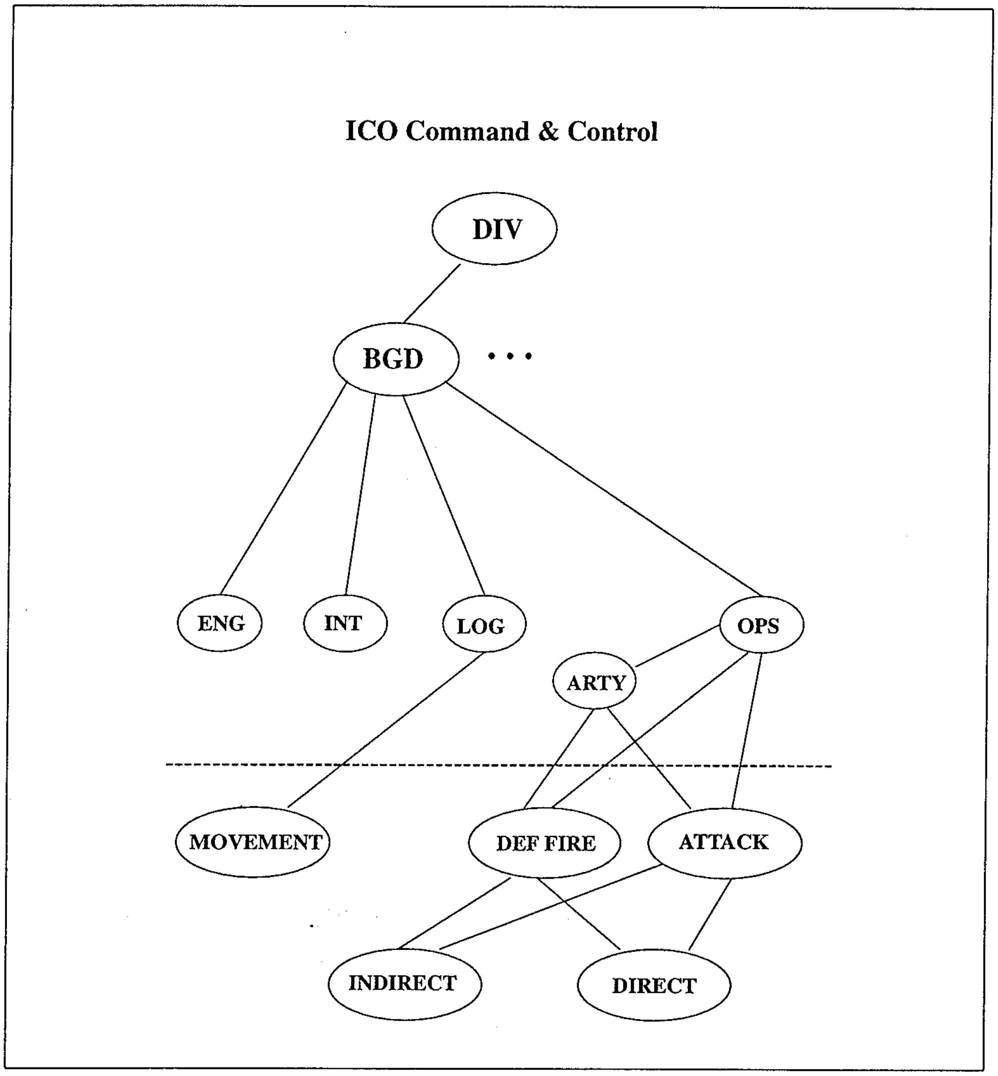


Figure 3: Command and Control flows down this graph. Division and Brigade command, and the subdomain experts are above the dashed line. Below the dashed line are lower level, WARFAC specific rules.

commander gives a movement order to Logistics. Logistics is responsible for carrying out the movements.

The current Intelligence subdomain expert does not request unit movements, so the Brigade commander has no conflicting requests to resolve. It is proposed that requests from multiple subdomain experts could be resolved by combination along the lines suggested in Ling and Rudd (9). They define formulae for combining numerical representations of expert opinions that may have stochastic dependencies.

3.5 Current Implementation

Nexpert permits an incremental development of the rule base. Thus, a working prototype with increasing degrees of sophistication could be developed. Since one of the motivations for building this system was that it should model human C^2 decision-making, the ICO had access only to data that a human player could acquire. For example, the locations of only those enemy units (i.e., the human's units) that would be visible on an Amiga map display. The ICO had no privileged access to WARFAC data structures. WARFAC specific knowledge and procedures were isolated as much as practicable, in the interest of maximizing the adaptability of the expert system to other environments.

The working prototype at the time of this document is an ICO with defensive capabilities. Rules for offensive moves (which take an initiative to attack and obtain a high-level objective) were not implemented, although this could be done in future versions. The scenario that has been used to test the ICO is the defence of a bridge against an attacking force. In the bridge defence scenario, the WFICO's forces start in defensive position around a bridge. The ICO rule set will attempt to maintain possession of the bridge. Briefly, it does this by holding the position of its units. If a unit is attacked and repelled from a position, the WFICO attempts to regain the position. Defensive fire is concentrated on attacking units that are deemed to pose the greatest threat. This threat level is measured by the cumulative amount of damage that has previously been inflicted on friendly forces by each attacking unit. If necessary, units will move into new positions to bring fire against the attacking units that pose the greatest threat. This movement is only done within constraints that maintain the units being moved in a defensive position about the bridge.

The initial set of knowledge structures in the ICO is quite small. This number grows dynamically over the course of a game, for example, as new enemy units are detected. The ICO currently contains approximately one hundred rules. Rule development was focused on the Operations subdomain expert. It contains the majority of the high level rules in the system. Most of the rules are lower level, game specific rules (below the dashed line in Figure 3) that support those higher ones. Figure 4 shows a partial set of the higher level rules contained in the Operations subdomain expert that are used to move defending units in response to a threat. The inferencing subtree containing the rules in Figure 4 is shown in Figure 5.

The ICO only uses partial knowledge of unit characteristics (e.g., it does not consider which weapons and sensors an enemy unit has), and does not yet incorporate terrain knowledge when it is considering movement. The addition of this knowledge could enhance the defensive abilities of the ICO, and would likely be essential for a sophisticated offensive capability. The FMP currently

```
RULE : Rule 67
If
                    ops.orders is "movement"
                    And there is evidence of ops_movement_init_threat
                    And there is no evidence of ops_movement_threat
Then ops_movement
                    is confirmed.


RULE : Rule 107
If
                    ops.orders is "movement"
                    And there is evidence of ops_movement_set_threat
                    And there is evidence of ops_movement_init_moving
                    And there is no evidence of ops_movement_moving
Then ops_movement_threat
                    is confirmed.
                    And Delete Object 'unit'\threat_unit_id\ |units_threat_c|
                    And Reset ops_movement_threat


RULE : Rule 102
If
                    ops.orders is "movement"
                    And there is evidence of ops_movement_set_moving
                    And there is evidence of ops_movement_move
Then ops_movement_moving
                    is confirmed.
                    And Delete Object 'unit'\move_unit_id\ |units_unmoved_c|
                    And Reset ops_movement_moving


RULE : Rule 100
If
                    ops.orders is "movement"
                    And there is evidence of ops_movement_calc_n
                    And there is evidence of ops_movement_calc_R
                    And there is evidence of ops_movement_calc_DCE
                    And there is evidence of ops_movement_calc_DUE
                    And DUE-R is greater than 0
                    And DCE-CEIL(R+R/n) is less than or equal to 0
                    And there is evidence of ops_movement_calc_T
Then ops_movement_move
                    is confirmed.
                    And Tx is assigned to 'unit'\move_unit_id\.wannabe_x
                    And Ty is assigned to 'unit'\move_unit_id\.wannabe_y
```

Figure 4: Some Battle Management Movement Rules

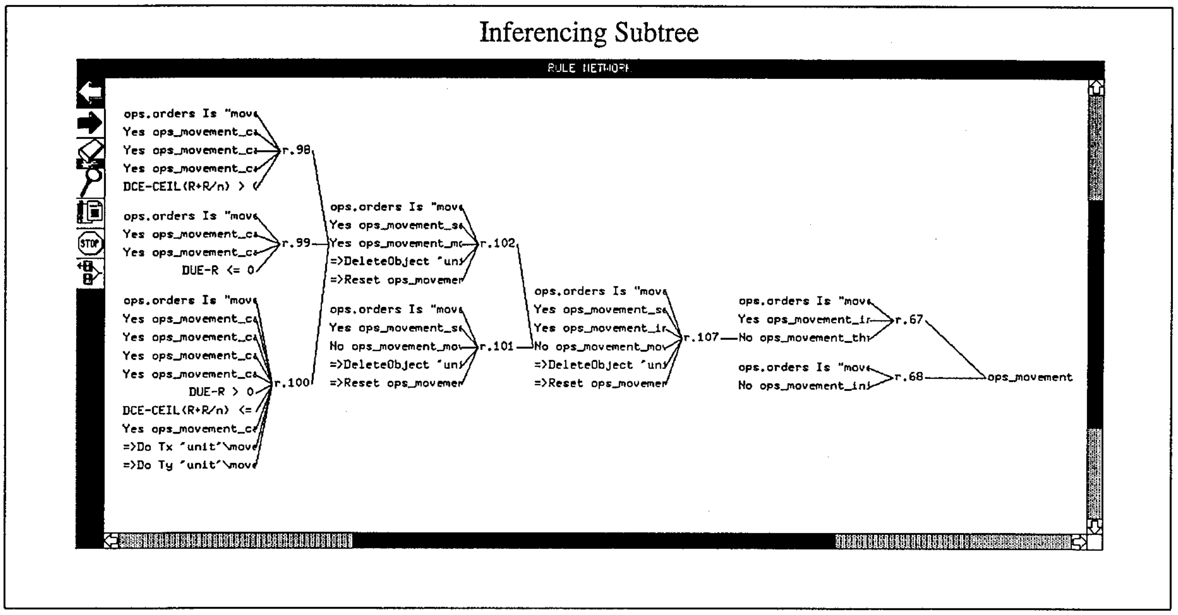


Figure 5: Partial Inferencing Subtree for rules in Figure 4.

accepts all types of messages that can be sent from the wargame[1]. Many of these messages contain potentially useful information that the ICO does not presently extract or take advantage of.

3.6 Evaluation

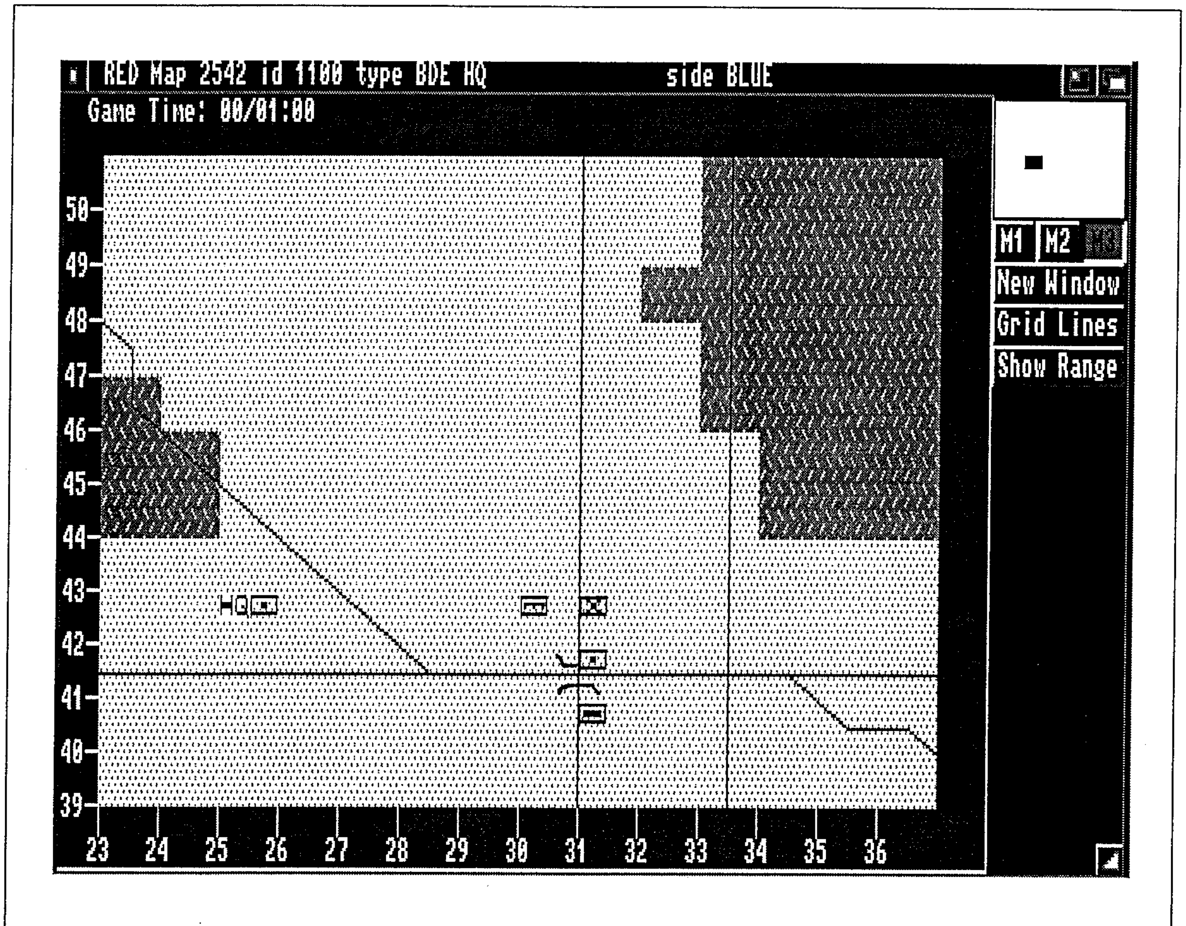
Even with a small base of partial world knowledge and very simple rules governing fire and movement, the ICO is able to respond well enough to enemy attacks to provide a reasonably challenging opponent to a sophisticated attacker.

Figure 6 shows a closeup of the ICO's units in their initial position around the bridge (The vertical line at map $x = 31$ is a river, shown in blue on the colour display, the other lines are roads, shown in black). When an enemy unit comes within sensing range, the ICO determines whether any of its units can be moved into position to fire upon the enemy unit, while still keeping the bridge within their range of fire.

Figure 7 shows the position from Figure 6 after the ICO has moved artillery units towards a threat, the enemy units off to the right. The artillery units have moved so the enemy units are just inside the rightmost limit of their range. If an enemy unit in another location becomes a greater threat, the artillery will be redeployed, provided it is not pinned down by enemy fire, and it is capable of moving. Attacking units are considered to be greater threats once they attack defending units of high value, such as a headquarters. The ICO prioritizes pre-emptive defensive fire to attack the highest threats first.

The level of play provided by the rules in the ICO is sufficient to require careful planning by the attacker to defeat the defending force. This has been accomplished by an experienced human player in about one hour of playing time, starting with a small force equal in size and capability to the defending

[1]Over one hundred message types.



Unit Type	Map Symbol
INFANTRY	
ARTILLERY	
TANK	
APC	

Figure 6: Initial defensive position.

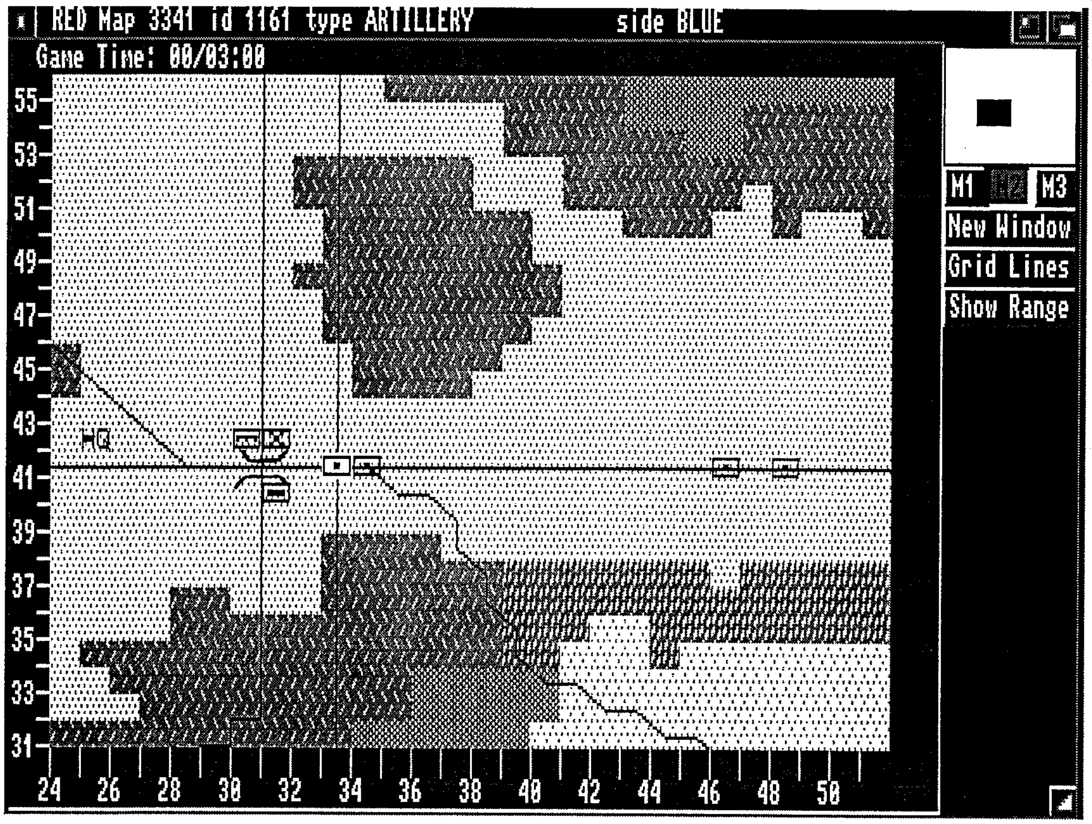


Figure 7: The position from Figure 6 after responding to the threat posed by the enemy units to the right.

force.

4 Nexpert Object as a tool

While Nexpert Object has proved an adequate tool for implementation of this project, some limitations in it have become evident. The WFICO was implemented using version 1 (6) of Nexpert Object. Features have been improved and added in version 2, but these limitations are still present. The implementation of object oriented knowledge structures is limited. Objects reside in a global name space, making encapsulation and data hiding awkward at best, and the structure that objects can have is fairly limited. These limitations can be mitigated by using external routines for representing and manipulating parts of the knowledge base.

The inferencing engine of Nexpert is excellent. The graphically based development environment provides advantages for overviewing and debugging small sets of rules. However, as the ICO grew larger than a few dozen rules and objects, the development tools rapidly became rather awkward to use. Moving through the entire knowledge base to find and edit specific rules and objects became a repetitive and time consuming task. While Nexpert does provide for splitting projects into multiple separate knowledge bases, this strategy causes forward reference problems and is best either used sparingly or avoided entirely. In comparison, all the generic expert system shells have limitations with respect to any specific project. These must be weighed against the costs inherent in developing a custom environment using a more general purpose programming language.

5 Uses of WFICO

5.1 Assessment of Human C^2 Decision Processes

A simulation of army warfare provides a medium for the study of decision processes involved in C^2. Clearly, the scope of thinking and behaviour that can be assessed will depend on the realism and level of detail of the simulation. The WARFAC and WFICO opponent provide a basis for the study of high-level decision processes; those processes that are concerned with the unfolding in space and time of a combat scenario. The WFICO, as a configurable opponent that will behave consistently in meeting a human challenger, permits a systematic study of these high-level decision processes. In this context the following questions could be addressed through experiment:

1. What is the relationship between the size of the forces and the time required to make effective use of them?
2. How do the decision processes differ between the deployment of forces and the use of those forces once they are deployed?
3. How can a "good" plan be recognized and evaluated?
4. How do time pressure, fatigue and other stress inducing factors affect the decision making processes?

While such questions may be easily posed, it becomes a challenge to determine how human performance might be measured to find reasonable answers. Thus, performance measurement becomes a major issue in designing a specific

experiment. In the current state of WARFAC, only a few performance measures are accessible: the number of turns of play it takes to reach an objective, the time taken by the human player in each game phase and the status of the forces (unit parameters such as strength). The physiological state of the player can also be monitored and associated with specific actions and game states.

Future work with WARFAC will involve the determination, for the particular experiment of interest, of performance measures that could be built into the system. These may be divided into two sets: outcome measures and process measures. Outcome measures could include points lost on either side, the total time taken, and whether the objective was (or was not) achieved. Such measures tend to be global in nature and are likely to be uninformative about the decisions that were made during the wargame.

Process measures, on the other hand, assess ongoing decision processes as the wargame progresses. These types of decisions will likely require measures to be computed and stored dynamically as they occur. Planning processes are often difficult to assess since they are largely non-verbalized mental activities. A medium for expressing and thus recording mental processes would help in understanding why particular decisions were made, as the plan provides context. A review of division-level command and control (10) discusses these issues and provides a basis for designing measures that could be of use with WARFAC.

5.2 Assisting the Human in a C^2 Decision Task

The WFICO has been designed to play against another player. However, provision was made in the design for adding a WARFAC advisor mode, where the expert system would suggest actions to a human player rather than playing directly. This allows the possibility of computer assisted game play. The WFICO would sit between the wargame and the human player at an Amiga workstation, and monitor game messages and player responses. Before performing an action, the player would turn to the WFICO to see what action it recommends. The player would be able to ask the WFICO why it would perform a particular action, before deciding whether to accept its recommendation. The WFICO would be able to detect when the player's response was different from its own. Differences could prompt a query to the player as to why the move was being made, and the answer could be recorded.

There is also the possibility of using the advisor mode in conjunction with machine learning techniques, to allow the WFICO to move its own decision model towards the behavior exhibited by the human player. This could be done by using knowledge of the conditions (state of the world) and the action performed to create a new rule for the knowledge base. A model of how a particular human player is making decisions could be built up in an incremental manner. The display capabilities of Nexpert would allow this model to be shown as a condition-action graph. The ability to create new rules for dynamic addition to a knowledge base is a substantive task as it requires that each new rule be checked for consistency with the existing knowledge base, and an appropriate place for its insertion must be determined.

The addition of a simple advisor mode would be primarily a matter of enhancing the existing user interface, which is intended as a debugging aid for the WFICO developer. A rudimentary explanation facility, allowing the player to scan backwards through the sequence of rules that led to the current recommended action, could be done easily using the built in explanation capabilities of Nexpert. To explain higher level motivations, and bypass long chains of trivial rules, a more elaborate explanation facility would have to be designed.

This would be a significant undertaking, and is an area of active research (11).

5.3 Testing Decision Making Rules

Like machine-based chess players, the WFICO could be played against itself. Of course, in its current implementation it is not well suited to this as it is a purely defensive player. However, if an offensive component were developed then it would be possible to test rules of decision making in machine on machine wargames. In such an environment, problems could be identified and decision processes refined in a cycle of development. Not only might this produce a better game player, but it could provide insights into more general C^2 decision processes. The validity of such an exercise would depend upon the realism of the scenario being simulated with WARFAC.

6 Acknowledgements

Bug fixes and minor enhancements to the wargame in support of the WFICO were done by Robert Manley. Jeff Hunter served as an expert human opponent for testing and evaluation of the WFICO.

7 Glossary

Backward Chaining Starting with a goal, and breaking it up into subgoals recursively to find a solution to the initial goal.

Declarative Knowledge Information that is expressed as a fact. E.g., The cow is purple. Declarative knowledge may be generated from procedural knowledge.

Expert System A computer program designed to solve problems in a restricted domain.

Expert System Shell A specialized programming language designed for implementing expert systems for arbitrary restricted domains.

Inference Engine A computer program that operates on sets of logical rules according to one or more control strategies, such as backwards chaining.

Knowledge Based System A computer program that works with declarative and procedural information to solve problems. Expert systems are often knowledge based.

Object Oriented Encapsulated classes and objects that are able to inherit properties from each other according to an inheritance hierarchy.

Procedural Knowledge Information that is represented as a rule or set of rules, that may be used to generate new rules and declarative knowledge.

Production System A computer program incorporating an inference engine and a set of rules and data. An expert system can be a knowledge based production system.

Subdomain Expert An expert system component with a domain of reasoning that is restricted to a specific area within the problem domain.

References

1. NTT Systems Inc. (1991) *SUSOPS WARFAC Users Guide* Contract # W7711-1-3457 to Defence and Civil Institute of Environmental Medicine, North York, Ont., Canada
2. BARR, A. and E.A. FEIGENBAUM. (1982) *The Handbook of Artificial Intelligence, Vol 2.* Stanford, CA: Heuristech.
3. Department of the Army, US, (1986) *Defensive Operations Doc # FM 100-5* Operations HQ, Department of the Army (US).
4. Department of National Defence (1985) *Land Formations in Battle, from: Land and Tactical Air Ops (Vol 1). Doc # B-GL-301-001/FP-001* Canadian Forces Publication.
5. CHORAFAS, D. N. (1990) *Knowledge Engineering* New York, NY: Van Nostrand Reinhold.
6. Neuron Data Inc. (1989) *Nexpert Object Fundamentals* Palo Alto, CA: Neuron Data Inc..
7. WINSTON, P.H. (1984) *Artificial Intelligence 2nd Ed.* Reading, MA: Addison-Wesley.
8. Department of National Defence (1984) *CFP The Army. Doc # B-GL-300-000/FP-000* Canadian Forces Publication.
9. LING, X. and W.G. RUDD. (1989) *Combining Opinions from Several Experts* Applied Artificial Intelligence, **3**, 439-452.
10. ENGEL, G.R. and M. TOWNSEND. (1991) *Division Command and Control Performance Measurement.* Contract # 01SE.W7711-7-7013 to Defence and Civil Institute of Environmental Medicine, North York, Ont., Canada
11. JACKSON, P. (1986) *Introduction to Expert Systems* Wokingham, England: Addison Wesley Publishing Company.

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g., Establishment sponsoring a contractor's report, or tasking agency, are entered in section 12.)

2. DOCUMENT SECURITY CLASSIFICATION (overall security classification of the document including special warning terms if applicable)
Unclassified

3. DOCUMENT TITLE (the complete document title as indicated on the title page. Its classification should be indicated be the appropriate abbreviation (S,C,R or U) in parentheses after the title.)
An Expert System Opponent for Wargaming

4. DESCRIPTIVE NOTES (the category of the document, e.g., technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

DCIEM Report

5. AUTHOR(S) (Last name, first name, middle initial. If military, show rank, e.g. Burns, Maj. Frank E.)

T.R.H. Cutmore and H.D. Gamble

6. DOCUMENT DATE (month and year of publication of document)	7.a. NO. OF PAGES (total containing information. Include Annexes, Appendices, etc.)	7.b. NO. OF REFS. (total cited in document)
April 1995	21	11

8.a. PROJECT OR GRANT NO. (if appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant)	8.b. CONTRACT NO. (if appropriate, the applicable number under which the document was written)

9.a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.)	9.b. OTHER DOCUMENT NO.(S) (any other numbers which may be assigned this document either by the originator or by the sponsor.)
DCIEM No. 93-54	DCIEM NO. 93-54

10. DOCUMENT AVAILABILITY (any limitation on further dissemination of the document, other than those imposed by security classification)

X	Unlimited distribution
	Distribution limited to defence departments and defence contractors; further distribution only as approved
	Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
	Distribution limited to government departments and agencies; further distribution only as approved
	Distribution limited to defence departments; further distribution only as approved
	Other

11. ANNOUNCEMENT AVAILABILITY (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (10.) However, where further distribution (beyond the audience specified in 10) is possible, a wider announcement audience may be selected.)

12. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.)

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual).

Planning and decision making for battle management are difficult and time-critical tasks. to facilitate the study of these cognitive processes in the laboraatory, an army wargame facility (WARFAC) has been developed at DCIEM. This report details the design and implementation of an extension to WARFAC, an expert system opponent. This extension will allow controlled experiments with a single human subject. Initial test results are presented and its possible uses as a research tool are outlined.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible, keywords should be selected from a published thesaurus, e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

DSIS DCD03
HFD 07/94